Rule-Based Estimation and Control of Formation Flying Spacecraft

Fred Y. Hadaegh*, Bryan H. Kang, and Daniel P. Scharf
Jet Propulsion Laboratory
California Institute of Technology
4800 Oak Grove Drive, Pasadena, CA 91109-8099

Abstract: This paper addresses some estimation and control problems specific to formation flying of spacecraft that can be approached with rule or logic based methodologies. For estimation, a decentralized and self-centered framework is formulated, and a rule-based strategy is applied to the problem that uses gain switching of the linear observer of the formation system. This particular strategy has the advantages of computational efficiency and fast filter convergence during formation initialization or formation member acquisition. For the control, a rule-based algorithm is applied to achieve synchronized deadband drift periods between spacecraft of desired length for scientific observation. The algorithm uses the deadband control of the Cassini spacecraft in concert with a straightforward equation relating the drift period to the peak drift magnitude. Rules are used to determine the necessary changes in drift period. Also, only position measurements and thruster firing magnitudes are used with no need for; velocity measurements and estimate of the disturbances. A two spacecraft simulation validate the efficacy of the algorithm.

Key Words: Formation-Flying, Rule-Based Control, Rule-Based Estimation, Distributed systems, Collaborative systems

Technical Area: T4

1. Introduction

Formation flying spacecraft refers to a set of spatially distributed spacecraft flying in formation with the capability of interacting and cooperating with one another. NASA's future Earth and space science missions involve formation flying of multiple coordinated spacecraft. Deployment of large numbers of low cost miniaturized spacecraft will enable to achieve a potentially powerful spaceborne system with flexibility to introduce new members to the formation for expansion, for upgrading technologies, or to replace a failed member. The collective behavior of all the spacecraft in the formation will determine the quality and the magnitude of the science return. The formation flying system must act collaboratively as a single collective unit, which represents a common system to perform a task. This revolutionary vision is in contrast to the old approach of deploying large and expensive multiple payload platforms.

In order to achieve this unique collective and collaborative characteristic in a formation estimation and control system, numerous technical challenges are foreseen. Most significantly, the complexity and interactive nature of the formation system calls for unconventional approaches in design that considers inter-spacecraft communications of states and measurements as well as commanding of formation-wide synchronized maneuvers. Thus multi-element (agent), collaborative objectives in this problem setting promise potential advantages in implementing less conventional approaches in spacecraft Guidance, Navigation and Control such as the intelligent systems approach (AI, rule-based, fuzzy logic etc.). The main objective of this paper is to present formation flying estimation and control problems that are well suited for rule-based (or logic-based) estimation and control approaches. For the problems defined, appropriate design

*Corresponding author, Email: Fred.Y.Hadaegh@jpl.nasa.gov, Tel: 818-354-8777, Fax: 818-393-0342

methodologies are provided and demonstrated to prove the concept.

2. Rule-Based Estimation

A rule-based estimation methodology for generic formation flying spacecraft is considered in the context of a decentralized and self-centered approach. The role of a formation estimator is to obtain the formation state estimates, which are necessitated by the formation control, through processing of collected inter-spacecraft sensor measurements. The formation estimator can be implemented in several different forms including centralized, decentralized and distributed, where each architecture requires its unique communication and implementation methods. The formation estimation architecture is decentralized if the formation lacks a formation-wide master filter. Each spacecraft has its own formation filter based on locally available information. A significant advantage of the decentralized filter over a centralized filter is that the formation estimation process can be made far more robust against a single point failure of formation members.

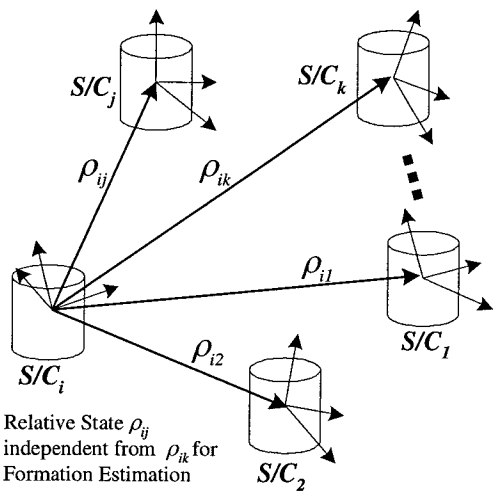


Figure 1: Self-centered representation of formation states

A decentralized estimation approach is to describe the formation system with self-centered perspective as shown in Figure 1. Each spacecraft describes the relative state variables with respect to itself in one to one manner. With this description, a failure of formation member does not impact other state variables during the propagation. Single fault estimation failure occurs only if the filter residing on a spacecraft fails. Therefore, a failed member or a disrupted inter-communication link can be gracefully tolerated by using the self-centered definition of the formation state vector (Figure 1). In addition, this self-centered approach retains optimality for other variant state descriptions that are obtained by a linear transformation of the self-centered state vector.

The decentralized and self-centered estimation methods are approached with several assumptions on the formation members. It is assumed that every spacecraft within the formation is similarly configured and equipped with its own processor, inertial sensors and restricted/limited field of view (FOV) relative position sensors. The inertial sensors may include gyros and accelerometers, and the restricted FOV sensors may include GPS-like sensors such as Autonomous Formation Flying (AFF) sensors [1] and laser or vision based sensors [2-4] e.g., a linear metrology sensor. The inter-spacecraft data can take two different forms, direct sensor measurement of the filter residing on a spacecraft, or communicated measurements and/or state estimates through inter-spacecraft communication network.

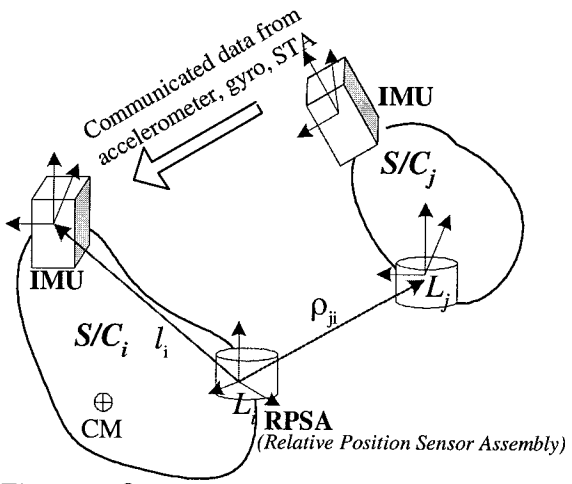


Figure 2: Observation link between S/C_i and S/C_j

2.1. Communication and Observation Link

Let's consider an observable link between ith and jth spacecraft (S/C_i and S/C_j) as shown in Figure 2. The relative motion between S/C_i and S/C_j at the relative position sensor assembly (RPSA) node is:

$$\begin{aligned} \ddot{\rho}_{ij} &= a_i^{JL} - a_j^{JL} \\ &= a_i^{JA} - M_{ij} a_j^{JA} \end{aligned}$$

$$
\begin{aligned}
&+\omega_i^{JB} \times \left(\omega_i^{JB} \times l_i\right) \\
&-M_{ij}\left(\omega_j^{JB} \times \left(\omega_j^{JB} \times l_j\right)\right) \\
&+\alpha_i^{JB} \times l_i - M_{ij}\left(\alpha_j^{JB} \times l_j\right) \quad (1)
\end{aligned}
$$

where a_j^{JL} is the acceleration of L_j node in inertial frame, a_j^{JA} is the acceleration of accelerometer node in inertial frame, M_{ij} is the directional cosine matrix from jth spacecraft to ith spacecraft, ω_j^{JB} and α_j^{JB} are the jth spacecraft body rate and angular acceleration w.r.t. inertial frame.

Substitute a_j^{JA} with the augmented IMU accelerometer model of:

$$
\begin{aligned}
a_{im}^{JA} &= a_i^{JA} + b_i + v_i \\
\dot{b}_i &= \eta_i \qquad (2)
\end{aligned}
$$

where b_i is an accelerometer bias and ν_i, η_i are white Gaussian noises. If the relative accelerometer bias is defined as: $b_{ij} = b_j - M_{ji} b_i$ and $b_i = b_{io}$ (assumed known because of the loss of observability without an attitude maneuver), then the propagation model becomes:

$$
\begin{aligned}
\begin{bmatrix} \dot{b}_{ij} \\ \dot{V}_{ij} \\ \dot{\rho}_{ij} \end{bmatrix} &= \begin{bmatrix} 0 & 0 & 0 \\ M_{ij} & 0 & 0 \\ 0 & I_{3x3} & 0 \end{bmatrix} \begin{bmatrix} b_{ij} \\ V_{ij} \\ \rho_{ij} \end{bmatrix} \\
&\quad + \begin{bmatrix} \eta \\ \nu \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ \zeta + \psi \\ 0 \end{bmatrix} \qquad (3)
\end{aligned}
$$

where:

$$
\begin{aligned}
\nu &= -\nu_i + M_{ij}\nu_j \\
\zeta &= a_{im}^{JA} - M_{ij} a_{jm}^{JA} + b_{io} \\
\psi &= \omega_i^{JB} \times \left(\omega_i^{JB} \times l_i\right) + \alpha_i^{JB} \times l_i \\
&\quad - M_{ij}[\omega_j^{JB} \times (\omega_j^{JB} \times l_j) + \alpha_j^{JB} \times l_j] \qquad (4)
\end{aligned}
$$

Note that the communicated data from jth spacecraft to ith spacecraft are $M_{Jj}, \omega_j^{JB}, \alpha_j^{JB}, l_j$ and a_{jm}^{JA} for the propagation and ρ_{lk} for any arbitrary l and k for the filter update.

2.2. Self-Centered Description of Formation System

The formation system of Figure 1 can be constructed by equation (3) for each inter-spacecraft observation link as follows. Equation (3) can be simplified and written as:

$$
\dot{x}_{ij} = A_{ij} x_{ij} + w_{ij} + u_{ij} \qquad (5)
$$

By defining the formation state X as:

$$
\begin{bmatrix} b_{1i} & V_{1i} & \rho_{1i} & \cdots & b_{ni} & V_{ni} & \rho_{ni} \end{bmatrix}^T, \qquad (6)
$$

the self-centered system of Figure 1 can be written as:

$$
\begin{aligned}
\dot{X} &= \begin{bmatrix} A_{1i} & 0 & 0 & 0 \\ 0 & A_{2i} & 0 & 0 \\ 0 & 0 & \ddots & 0 \\ 0 & 0 & 0 & A_{ni} \end{bmatrix} X + \begin{bmatrix} u_{1i} \\ u_{2i} \\ \vdots \\ u_{ni} \end{bmatrix} \\
&= AX + U \qquad (7)
\end{aligned}
$$

And equation (7) can be discretized to obtain the propagation equation of:

$$
\hat{X}_{k+1/k} = \Phi_k \hat{X}_{k/k} + \Gamma_k U_k \qquad (8)
$$

As shown by the equation (7), the propagation of the formation filter is isolated and independent for each observable links. Based on this formulation, body rate, angular acceleration, attitude and accelerometer output should be communicated over to the ith spacecraft in order to perform propagations. In the absence of communication, propagation can just continue with last data available on those variables.

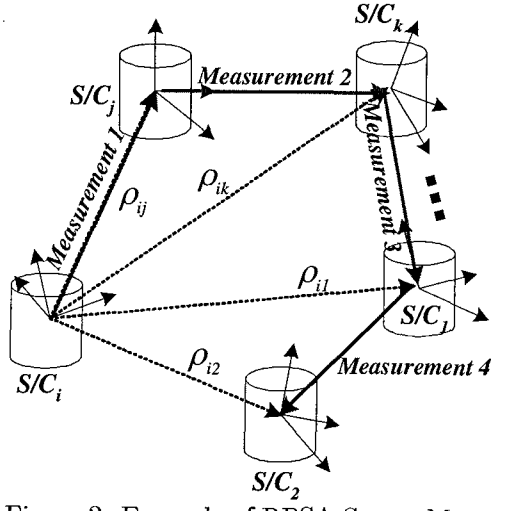


Figure 2: Example of RPSA Sensor Measurements

2.3. Relative Position Sensor Model

For each RPSA measurement (see Figure 3) the finite FOV observation can be generalized with following nonlinear bearing and magnitude measurement equations.

$$\begin{aligned} |\rho_{ij}| &= \sqrt{\rho_x^2 + \rho_y^2 + \rho_z^2} = h_1(x) \\ \gamma &= \cos^{-1}\left(\frac{\rho_z}{|\rho|}\right) = h_2(x) \\ \beta &= \tan^{-1}\left(\frac{\rho_y}{\rho_x}\right) = h_3(x) \\ |\dot{\rho}_{ij}| &= \sqrt{V_x^2 + V_y^2 + V_z^2} = h_4(x) \end{aligned} \tag{9}$$

Equation (12) is linearized about the state estimate to calculate the observation C matrix, and the finite FOV condition implies that the measurement is available only if $|\gamma| \leq \gamma_o$ and $|\beta| \leq \beta_o$. For example, if a communicated measurement of the measurement 2 is available at the filter hosting spacecraft i, then the observation equation becomes:

Since measurement 2 is ρ_{jk},

$$\begin{aligned} \rho_{jk} &= \rho_{ik} - \rho_{ij} \\ &= [0 \cdots : \ 0 \ \ 0 \ \ -1 : \ 0 \ \ 0 \ \ 1 : \ \cdots 0]X \\ &= C_{jk}X \end{aligned} \tag{10}$$

Thus the individual measurement update can be written as:

$$\hat{X}^+ = \hat{X}^- + H_{jk}\left(Z_{jkm} - C_{jk}\hat{X}^-\right) \tag{11}$$

2.4. Rule-Based Linear Observer Architecture

Given the propagation and observation equations (??) given in the previous section, a formation filter can be designed. Here, a simple rule based observer is considered for computational efficiency and filter robustness. If the filter is designed as a Kalman filter, then the large dimensional covariance matrix of Equation (6) must be propagated during every filter cycle by solving the Riccati equation onboard. As the formation dimension increases, the computational burden can be significant. On the other hand, if a linearized constant gain filter is implemented with the steady state gain, then the filter response can be too slow to handle initial acquisition of a formation member (or returning from partial outage). Therefore, a rule-based estimation scheme is approached as shown in Figure 4. Several linearized gains are calculated a priori based on each mode of operation, where these gains are then switched based on a simple rule.

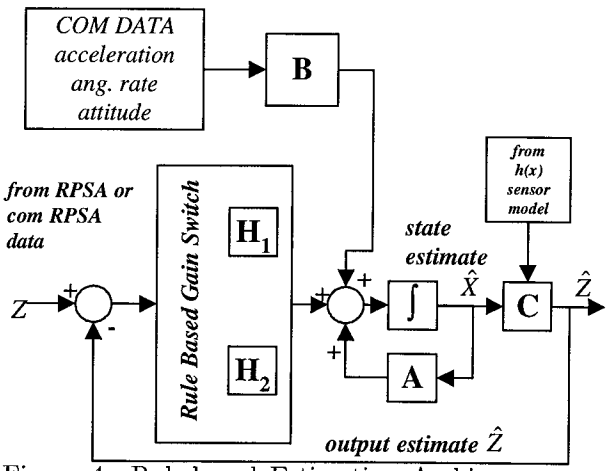


Figure 4: Rule-based Estimation Architecture

Rule-Based Gain Switching Algorithm

Step 1: For each measurement to be updated, calculate the measurement residuals Z_e as:

$$Z_e = Z_{ijm} - C_{ij}\hat{X} \tag{12}$$

where Z_{ijm} is the individual local measurements.

Step 2: Apply appropriate gains for different measurement residuals. By using the precomputed filter gains for the formation member capture and steady state, the filter gain will be switch by using following rule.

If $Z_e > Z_{switch}$ then

Use the large capture gain $H_{ij}^{capture}$ for the state update as:

$$\hat{X}^+ = \hat{X}^- + H_{ij}^{capture}(Z_{ijm} - C_{ij}\hat{X}^-)$$

else

Use the steady state gain H_{ij}^{ss} for the state update as:

$$\hat{X}^+ = \hat{X}^- + H_{ij}^{ss}(Z_{ijm} - C_{ij}\hat{X}^-)$$

endif

Step 3: Proceed with additional measurement updates or propagate to next filter cycle.

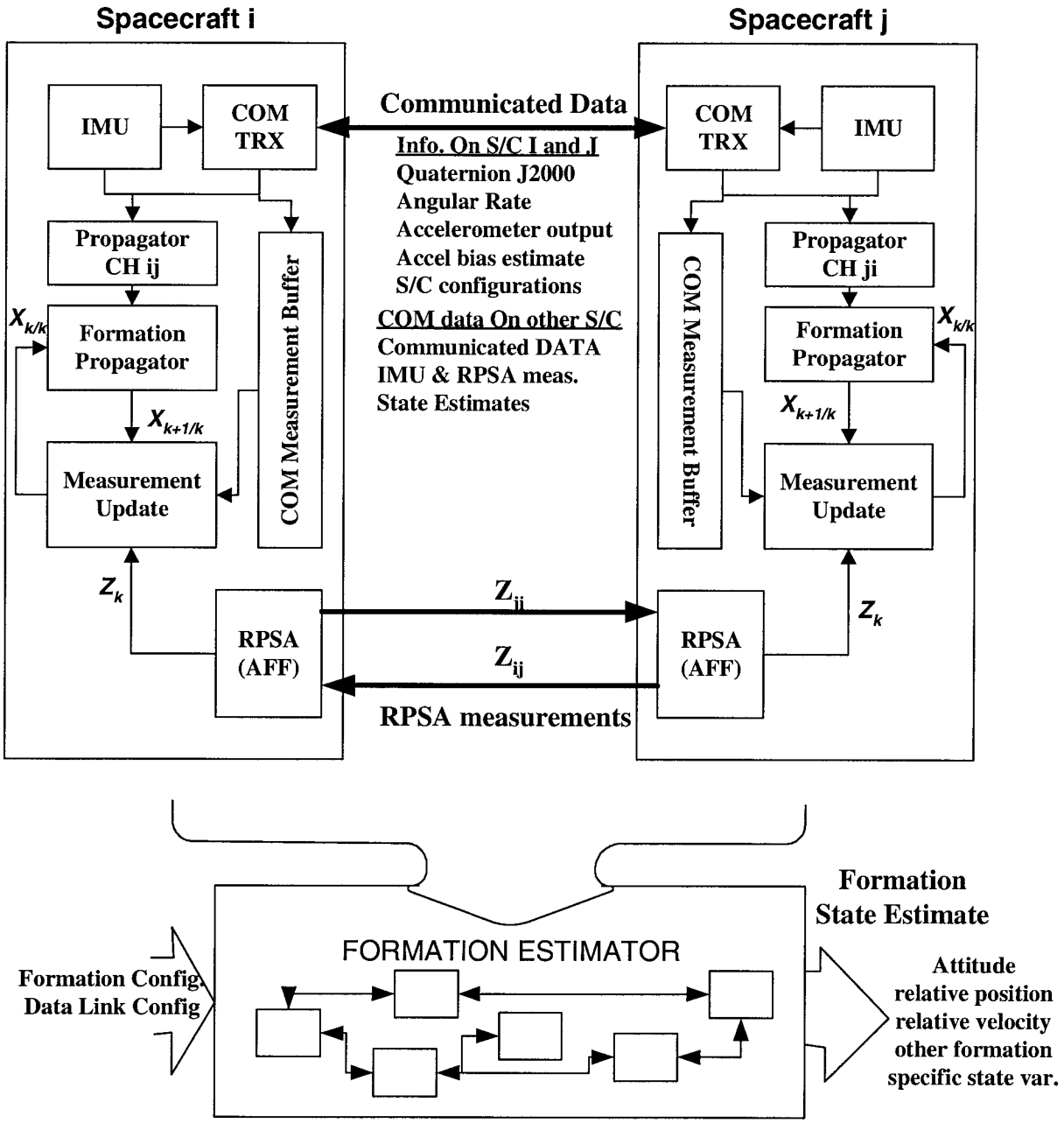


Figure 5: A decentralized estimation architecture

3. Rule-Based Control

An important use for a spacecraft formation is Michelson stellar interferometry; both astrometric and synthetic aperture imaging applications are possible [6, 7]. Present implementations of formation Michelson interferometry require spacecraft in the formation to track specified positions and attitudes relative to one another. In [6], Wang and Hadaegh develop continuous control laws to achieve these goals under the assumption that each spacecraft is maneuvering with respect to a "leader" spacecraft whose actions are independent of the "following" spacecraft (the Leader/Follower paradigm).

Another aspect of the stellar interferometry problem is that the quality of the interference fringes is very sensitive to control actuation. For NASA's Starlight mission, e.g., a two spacecraft formation for stellar interferometry, the spacecraft are only equipped with thrusters for position and attitude control. A requirement is that while light is being interfered, no thrusters may fire; the resulting vibration would ruin the interference fringes. In particular, the spacecraft must drift together, remaining inside their position and attitude deadbands for 27 seconds, and then have 3 seconds to fire their thrusters, and they must drift again. Their deadband drift periods (or simply "periods") must be *synchronized.* Each spacecraft has inertial pointing requirements and one spacecraft is responsible for maintaining relative position requirements. Therefore, for Starlight, periods about 9 axes must be synchronized ("synched").

The axes are assumed to be decoupled, double integrators with a constant disturbance acting. That is, each axis can be represented by the equation $\ddot{x}_i = a_i$, where a_i is constant and represents a solar pressure (torque) and x_i is a translational or rotational position. The a_i's need not be equal. While the a_i will depend on attitude and position, for small deadbands they will be nearly constant. The control actuator is assumed to be an ideal impulsive thruster. That is, if a thruster fires at time t with magnitude W, $\dot{x}_i(t_+) = \dot{x}_i(t_-) + W$.

Two types of deadbands may be considered. In the first, the velocity departing the reference edge (see Figure 6.a) is such that the disturbance pushes the spacecraft back to the reference edge before hitting the opposite edge. These will be called one-sided deadbands. In the second type, the velocity is such that the trajectory hits the other side of the deadband and a thruster firing (burn) is required to reverse the direction of the spacecraft. These will be called double-sided deadbands. Since quantization effects are not included here, only single-sided deadbands will be considered (the burn can always be made small enough so that the disturbance has time to act).

We are utimately interested in the ability to synchronize multiple single- and double-sided deadbands with quantization effects. This section is focusing on synchronizing two single-sided deadbands, though this is easily extendable to an arbitrary number of single-sided deadbands.

In Leader/Follower (L/F), follower spacecraft follows trajectories based on the motion of a leader spacecraft and the leader spacecraft acts independently of the followers. As such, inter-spacecraft feedback is removed and the formation is simpler to analyze.

The following algorithm to synchronize drift periods relies on the ability to achieve a specified drift period, T^i, where $i = L$, F or D, for Leader, Follower or Desired spacecraft respectively. With this capability, rules are used to choose the T^i. Echoing the discussion of deadbands by Breckenridge [5], the velocity departing the reference edge and the magnitude of the disturbance are sufficient to characterize the deadband. Considering the kth drift period, let $\dot{x}_k$ be the velocity departing the reference edge. The trajectory will return to the reference edge with velocity $-\dot{x}_k$. If a burn of magnitude W_k (recall this is an impulse) is performed, the next departure velocity is $\dot{x}_{k+1} = -\dot{x}_k + W_k$. It can be shown that $\dot{x} = -a_i T/2$ gives the drift period T, so $W_k = -\dot{x}_k - a_i T/2$ is the burn that is needed.

This method requires position and velocity measurements and an estimate of a_i. According to [9], both velocity measurements and disturbance estimates can be unreliable. Instead of this algorithm for specifying W_k, the Cassini Deadband Tuning Algorithm (or more simply the Cassini Algorithm) is used [9]. The Cassini Algorithm only uses position measurements and burn lengths and it will be discussed more fully subsequently.

Summarizing, the synchronization algorithm uses straightforward rules to determine the next T^L and T^F, and then the Cassini Algorithm is used to achieve these drift periods. The rules used will depend on the control architecture chosen.

The rules for the drift period synchronization algorithm are predicated on the rule-based rotational synchronization algorithm of Wang and

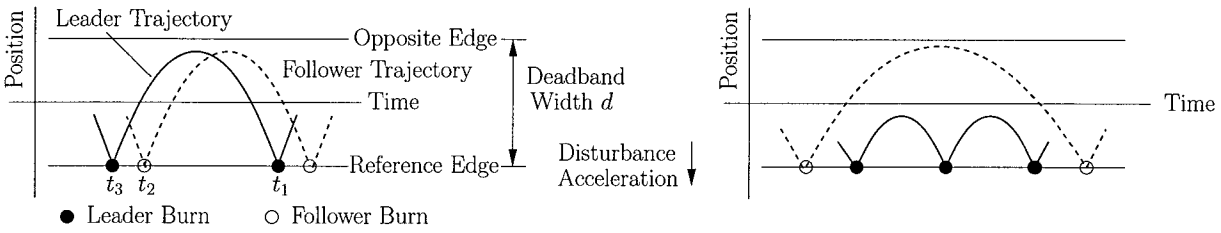


Figure 6: Deadband Geometry and Corresponding Burns

Hadaegh [8]. They are synchronizing the rotation of two spacecraft with an omnidirectional sensor and a directional laser; the only data are the times that the laser of either spacecraft hits the other's sensor. Also using the L/F paradigm, the follower first matches its period to the that of the leader and then slowly speeds up or slows down over multiple rotations to achieve the desired phasing. Likewise, for drift period synchronization, the follower first matches its period to the that of the leader and then performs a synchronization burn to match burn times. There is an additional pre-step in which the leader achieves the desired drift period before the follower tries to match periods.

4. Synchronization Algorithm

4.1. Determining In/Out of Synch from Burn Times

As a prerequisite for synchronization, the periods must be roughly the same. We assume that the follower has achieved the desired period, T^D, to within a percentage tolerance η. That is, $|T^L - T^D| \leq \eta T^D$, where T^L is the period of the leader spacecraft (time between burns). The periods of the leader and the follower are considered matched when the follower's period is within the percentage tolerance ρ of T^L. That is, $|T^F - T^L| \leq \rho T^L$, where T^F is the period of the follower spacecraft. Finally, the matched periods are considered in synch when the time difference between "corresponding" (to be defined in a moment) leader and follower burns is less than γT^D, where γ is another percentage tolerance.

The concept of corresponding burns brings out some subtleties in the problem formulation. The leader burn corresponding to a follower's burn is the nearest leader burn in time to the time of the follower's burn (and vice versa). In Figure 6.a, the corresponding leader burn can be found by taking the minimum of $t_1 - t_2$ and $t_2 - t_3$.

The case shown in Figure 6.b must also be considered. If two more leader (follower) periods can fit inside one follower (leader) period, then it is not possible to associate burns and the concept of being in synchronization (in synch) becomes nonsensical. Requiring $\rho < 1/2$ avoids this problem.

Also, for the definition of in synch to be self-consistent, it is required that when the leader and follower burns coincide (exactly in synch) and the ρ tolerance is met, then the next burns must also be considered in synch. Otherwise, no two periods can ever be in synch. The most that the times of the next burns of the leader and the follower can differ by when the previous burns were simultaneous is ρT^L. For these burns to be considered in synch, $\rho T^L < \gamma T^D$, which means γ must be greater than $\rho(1+\eta)$.

Let a burn sequence be the history of burns with L signifying a leader burn and F a follower burn. For example, LLF is two leader burns followed by a follower burn, the follower's burn being the most recent. By requiring $\rho < 1/2$, at most two identical symbols may appear in a row in a burn sequence. For example, LLLF and LFFF can not happen.

It is not possible to determine if periods are out of phase by differencing two burn times. In Figure 6.a, if t_1 and t_2 are differenced it may appear that the periods are out of synch, even though $t_2 - t_3 < \gamma T^D$. Therefore, a sequence of at least three burns is needed to determine whether or not the periods are in synch.

Considering burn sequences of length three, there are six possibilities (recall LLL and FFF are not possible). Four of the possibilities (LLF, FLL, FFL and LFF) imply a fourth burn. For example, since at most two symbols may appear in a row, LLF implies FLLF. That is, for those four possibilities, one period is bracketed by another. The burns may differ by at most $\rho T^L/2$, which is less than γT^D. Therefore, those four burn sequences imply that the periods are in synch.

LFL and FLF remain. A burn sequence of LFL is sufficient to determine if the periods are in or out of synch, and a synch burn can be performed if necessary. For FLF, the same can be said, but from an implementation point of view, the period of the leader, T^L, is not known. So nothing is done. However, FLF becomes either FLFL, and the subsequence LFL is obtained, or it becomes FLFF, which implies that the periods are in synch (FLFF can only become FLFFL).

Therefore, three requirements must be met for a synchronization burn to be considered: 1) T^L must be within tolerance, 2) T^F must be within tolerance and 3) the last three burns must be LFL.

4.2. Cassini Deadband Tuning Algorithm

Each spacecraft is matching periods and, during synchronization, a burn must be calculated by the follower to obtain a period which will cause the next follower burn to coincide with a burn of the leader. The Cassini Algorithm is part of the solution.

The deadband tuning method of the Cassini Attitude Control System is detailed in [9]. First, the deadband is specified in terms of the desired peak from the reference edge, L (see Figure 7.a). With reference to the figure, the thruster firing, W_{k+1} to achieve a peak height of L can be calculated using only L_1, L_2 and W_k. The formula from [9], page 16, is

$$W_{k+1} = W_k \frac{\sqrt{L_2} + \sqrt{L}}{\sqrt{L_1} + \sqrt{L_2}} \quad . \tag{13}$$

In particular, note that neither a velocity measurement nor an estimate of the disturbance is needed, only position measurements and the magnitude of the last burn.

4.3. Drift Period Tuning and Synchronization

Given the last drift peak values and the last burn magnitude, (13) can be used to obtain a desired L. We now consider how to choose L. Using the simple dynamics, the drift time, T, can be related to the peak height, L, and the disturbance acceleration, a (where the subscript i notation is dropped), by

$$L = \frac{|a|}{2}\left(\frac{T}{2}\right)^2 . \tag{14}$$

To change the period by ΔT requires changing the height by ΔL. Rewriting (14) with $T + \Delta T$ and $L + \Delta L$ and dividing both sides by L yields

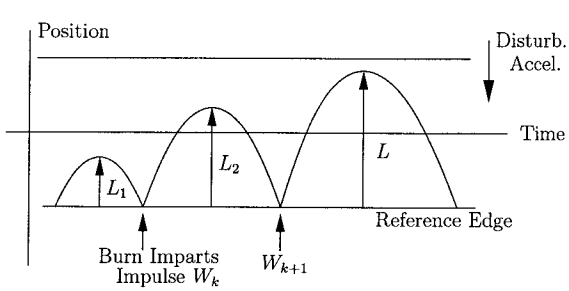
$$\frac{\Delta L}{L} = 2\frac{\Delta T}{T} + \left(\frac{\Delta T}{T}\right)^2 , \tag{15}$$

which relates a percentage change in L to a percentage change in T. Note that ΔL depends only on T and the desired change, ΔT; a is not needed.

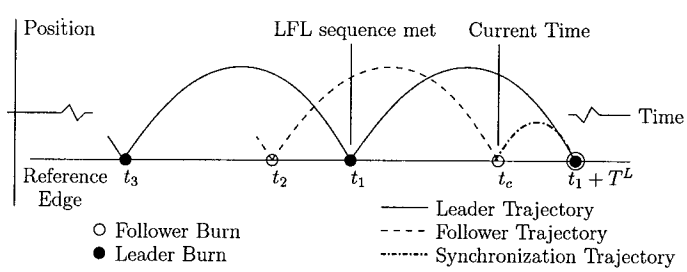
To match periods, the follower records the burn times of the leader to obtain the leader's period, then knowing its own period it calculates the necessary ΔT. Using (15), the new L for the follower is calculated, and then (13) gives the new burn magnitude.

To synch periods, consider Figure 7.b. The drift period of the leader, $T^L = t_1 - t_3$, and the follower, $T^F = t_c - t_4$, can be calculated from the firing times, t_i. The next leader burn is at $t_1 + T^L$. To match the burns, the follower will drift $T^S = (t_1 + T^L) - t_c$ seconds after its next burn. Therefore,

$$\begin{aligned} \Delta T &= T^S - T^F \\ &= (t_1 + T^L) - t_c - (t_c - t_2) \\ &= (t_2 - t_3) - 2(t_c - t_c). \end{aligned}$$



a) Cassini Deadband Tuning Algorithm



b) Calculation of ΔT for Synchronization

Figure 7: Parts of Synchronization Algorithm

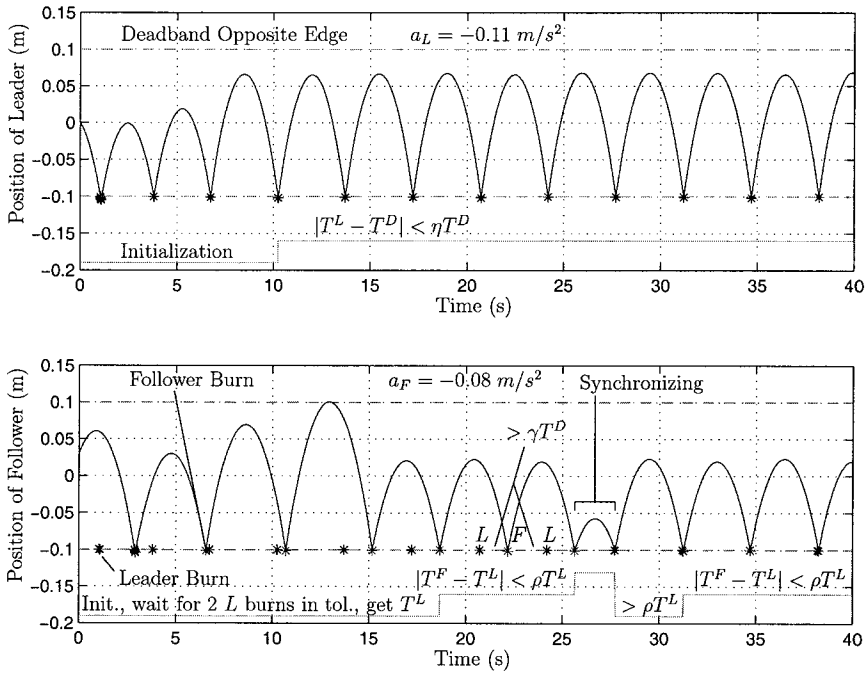


Figure 8: Simulation of Full Deadband Synchronization Algorithm

Again, (15) is used to calculate ΔL and then (13) gives the new burn magnitude.

5. Simulation

Figure 8 shows a simulation of the algorithm. The values are: $\eta = \rho = 0.03$, $\gamma = 0.06$, $T^D = 3.5$ seconds, $\lambda = 0.1$, $d = 0.2$ m, $a_L = -0.11 m/s^2$ and $a_F = -0.08$ m/s^2. The sample time is 0.01 seconds. The upper plot is of the leader's position, with the deadbands as dashed lines and the burns as black stars. The grey line at the very bottom of the upper plot is the state of the leader: when low, the drift period is not within the ηT^D tolerance and when high, it is within tolerance.

The lower plot is of the follower's position. In addition to its burns, shown as grey stars, the leader's burns (black stars) are also superimposed. The follower state has three levels: low, middle and high. Low signifies that the follower's drift period is not within the ρT^L tolerance. Recall the follower waits until the leader has two burns within tolerance before attempting to match the leader's drift period: the one hump which just touches the upper deadband limit is the follower maximizing its drift (the first three humps are part of the initialization routine). Once the leader is within tolerance, the follower matches (the next hump after the humps which just touch the upper deadband). Then, after the follower matches (signified by the medium height of the follower's state line which is the grey line at the bottom of the lower plot), synchronization can occur. At approximately 26 seconds, all the conditions for a synchronization burn are met: both spacecraft's drift periods are within their respective tolerances, the last three burns are LFL, and the burns right before the follower's state goes high (signifying a synchronization burn) are separated by more than γT^D. Note that the "steady-state" deadband cycles of the leader and follower are of different heights; the period is the same but since the disturbances are different, the values of L differ.

6. Conclusions

The rule-based estimation methodology is described. By adapting the decentralized and self-centered approach, the estimation method is fault tolerant from communication dropouts. The rule-based approach allows realization of a simple estimation algorithm with capability to handle dynamic recapture.

A rule-based, Leader/Follower paradigm, single-sided deadband drift period synchronization algorithm has been presented as well as simulation results showing its efficacy. It combines the underlying ideas of the rotation synchronization algorithm of Wang and Hadaegh with the Cassini Algorithm to synchronize deadbands without needing to estimate the disturbance ac-

celeration, a, nor does it require velocity measurements. There are a number of tolerances, but they are easily set. This method can be used to synchronize axes on one spacecraft as well as axes on different spacecraft. The next step in developing this algorithm is including quantization effects and then synchronizing single-sided and double-sided deadbands can be addressed.

7. Acknowledgements

The work described in this paper was carried out at the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration.

8. References

[1] Purcell, G.; Kuang, D.; Lichten, S.; Wu, S. and Young, L.; "Autonomous Formation Flyer (AFF) Sensor Technology Development", Feb 4-8, 1998, 21st AAS Guidance and Control Conference.

[2] Hirobumi, S.; Tatsuaki, H.; Kenji, K.; and Hiroshi, G.; "Micro-scanning laser range finders and position-attitude determination for formation flight", Annual AIAA/Utah State University Conference on Small Satellites, 13th, Logan, UT, Aug. 23-26, 1999, Report #: SSC99-VI-1

[3] Alonso, R.; Crassidis, J.; and Junkins, J.L.; "Vision-based relative navigation for formation flying of spacecraft", AIAA Guidance, Navigation, and Control Conference and Exhibit, Denver, CO, Aug. 14-17, 2000, AIAA Paper 2000-4439

[4] Arambel, P. O.; Rago, C.; Seereeram, S. and Mehra, R. K.; " Fault Tolerant Autonomous Controls for Autonomous Formation Flying Spacecraft using Interacting Multiple Models and Adaptive Nonlinear Predictive controls", NASA SBIR Phase II Final Report, Contract NAS 3-99130, Scientific Systems Company. March 11, 2001

[5] W.G. Breckenridge, "Limit cycle behavior in the presence of a constant disturbance." Jet Propulsion Laboratory Internal Document, Engineering Memo No. 343-205, 4800 Oak Grove Dr., Pasadena, CA 91109, 1974.

[6] P.K.C. Wang and F.Y. Hadaegh, "Coordination and control of multiple microspacecraft moving in formation," *J. of Astronautical Sciences*, vol. 44(3), pp. 315–355, 1996.

[7] P.K.C. Wang, F.Y. Hadaegh, and K. Lau, "Synchronized formation rotation and attitude control of multiple free-flying spacecraft," *J. Guid., Contr., and Dynam.*, vol. 22(1), pp. 28–35, 1999.

[8] P.K.C. Wang, J. Yee, and F.Y. Hadaegh, "Synchronized rotation of multiple autonomous spacecraft with rule-based controls, experimental study," *J. Guid., Contr., and Dynam.*, vol. 24(2), pp. 352–359, 2001.

[9] E. Wong, ed., "Cassini Project Control Analysis Book, vol. II." Jet Propulsion Laboratory Internal Document, Document No. JPL D-9683 PD 699-410, 4800 Oak Grove Dr., Pasadena, CA 91109, pgs. 15-18, 1998.

[10] B.J. Young, R.W. Beard, and J.M. Kelsey, "A control scheme for improving multi-vehicle formation maneuvers," in *Proc. American Cont. Conf.*, (Arlington, VA), pp. 704–709, 2001.